

**IN THE CLAIMS:**

*Set forth below in ascending order, with status identifiers, is a complete listing of all claims currently under examination. Changes to any amended claims are indicated by strikethrough and underlining. This listing also reflects any cancellation and/or addition of claims.*

- 1-25 (Cancelled)
26. (Original) A graphics pipeline system with an integrated masking operation, comprising:
- (a) a transform module adapted for being coupled to a buffer to receive graphics data therefrom, the transform module being positioned on a single semiconductor platform for transforming the graphics data from a first space to a second space; and
  - (b) a lighting module coupled to the transform module and positioned on the same single semiconductor platform as the transform module for performing lighting operations on the graphics data received from the transform module;
  - (c) wherein a masking operation is performed on the same single semiconductor platform as the transform module and the lighting module.
27. (Original) The system as recited in claim 26, wherein the masking operation is carried out utilizing a write mask.
28. (Original) The system as recited in claim 27, wherein the write mask includes a 2-bit write mask.
29. (Original) The system as recited in claim 28, wherein the write mask is adapted to provide control to the word level.
30. (Original) The system as recited in claim 26, wherein the masking operation masks a write operation to a predetermined register component.
31. (Original) The system as recited in claim 26, wherein the masking operation is carried out on a bit of a control vector for allowing analysis of remaining bits of the control vector.

32. (Original) The system as recited in claim 31, wherein the masking operation is capable of being used to convert vector graphics data to scalar graphics data.
33. (Original) The system as recited in claim 26, wherein the masking operation includes a color masking operation.
34. (Original) The system as recited in claim 33, wherein the masking operation involves an ambient mask, a diffuse mask, and a specular mask.
35. (Original) The system as recited in claim 26, and further comprising a rasterizer coupled to the lighting module for rendering the graphics data received from the lighting module, wherein the rasterizer is positioned on the same single semiconductor platform as the transform module and lighting module.
36. (Original) A method for graphics processing, comprising:
  - (a) transforming graphics data from a first space to a second space;
  - (b) lighting the graphics data; and
  - (c) performing a masking operation on the graphics data;
  - (d) wherein the graphics data is transformed and lighted, and the masking operation is performed on a single semiconductor platform.
37. (Original) The method as recited in claim 36, wherein the masking operation is carried out utilizing a write mask.
38. (Original) The method as recited in claim 37, wherein the write mask includes a 2-bit write mask.
39. (Original) The method as recited in claim 38, wherein the write mask is adapted to provide control to the word level.
40. (Original) The method as recited in claim 36, wherein the masking operation masks a write operation to a predetermined register component.

41. (Original) The method as recited in claim 36, wherein the masking operation is carried out on a bit of a control vector for allowing analysis of remaining bits of the control vector.
42. (Original) The method as recited in claim 41, wherein the masking operation is capable of being used to convert vector graphics data to scalar graphics data.
43. (Original) The method as recited in claim 36, wherein the masking operation includes a color masking operation.
44. (Original) The method as recited in claim 43, wherein the masking operation involves an ambient mask, a diffuse mask, and a specular mask.
45. (Original) The method as recited in claim 36, and further comprising rendering the graphics data, wherein the graphics data is rendered on the single semiconductor platform.
46. (Original) A computer program product for graphics processing, comprising:
  - (a) computer code for transforming graphics data from a first space to a second space;
  - (b) computer code for lighting the graphics data; and
  - (c) computer code for performing a masking operation on the graphics data;
  - (d) wherein the graphics data is transformed and lighted, and the masking operation is performed on a single semiconductor platform.
47. (Original) A system for graphics processing, comprising:
  - (a) means for transforming graphics data from a first space to a second space;
  - (b) means for lighting the graphics data; and
  - (c) means for performing a masking operation on the graphics data;
  - (d) wherein the graphics data is transformed and lighted, and the masking operation is performed on a single semiconductor platform.

48. (Previously added) A graphics pipeline system with an integrated masking operation, comprising:
- a transform module positioned on a single semiconductor platform for transforming graphics data;
  - a lighting module positioned on the same single semiconductor platform as the transform module, the lighting module being for performing lighting operations on the graphics data;
  - a set-up module positioned on the same single semiconductor platform as the transform module and the lighting module, the set-up module being for setting up the graphics data; and
  - a rendering module positioned on the same single semiconductor platform as the transform module, the lighting module, and the set-up module, the rendering module being for rendering the graphics data;
- wherein a masking operation is capable of being performed utilizing the single semiconductor platform.
49. (Previously added) The system as recited in claim 48, wherein the masking operation includes a 2-bit write mask for providing control to a word level of a buffer address.
50. (Previously added) The system as recited in claim 48, wherein the masking operation includes individually masking a write operation to at least one register component such that unmasked components are taken from the register and masked components are bypassed.
51. (Previously added) The system as recited in claim 48, wherein the masking operation masks an enabled bit of a control vector for allowing analysis of all remaining bits after execution of the masking operation.
52. (Previously added) The system as recited in claim 51, wherein the masking operation masks vector vertex data for converting the vector vertex data to scalar vertex data.
53. (Previously added) The system as recited in claim 48, wherein the masking operation is associated with an ambient attribute.

54. (Previously added) The system as recited in claim 48, wherein the masking operation is associated with a diffuse attribute.
55. (Previously added) The system as recited in claim 48, wherein the masking operation is associated with a specular attribute.
56. (Previously added) The system as recited in claim 48, wherein the single semiconductor platform is adapted for coupling to a central processing unit for receiving instructions therefrom.
57. (Previously added) A method for graphics processing, comprising:  
transforming graphics data from a first space to a second space;  
lighting the graphics data;  
performing a masking operation on the graphics data;  
setting up the graphics data; and  
rendering the graphics data;  
wherein the graphics data is set up, transformed and lighted, and the masking operation is performed, on a single semiconductor platform.
58. (Previously added) The method as recited in claim 57, wherein the masking operation includes a 2-bit write mask for providing control to a word level of a buffer address.
59. (Previously added) The method as recited in claim 57, wherein the masking operation includes individually masking a write operation to at least one register component such that unmasked components are taken from the register and masked components are bypassed.
60. (Previously added) The method as recited in claim 57, wherein the masking operation masks an enabled bit of a control vector for allowing analysis of all remaining bits after execution of the masking operation.
61. (Previously added) The method as recited in claim 60, wherein the masking operation masks vector vertex data for converting the vector vertex data to scalar vertex data.

62. (Previously added) The method as recited in claim 57, wherein the masking operation is associated with an ambient attribute.
63. (Previously added) The method as recited in claim 57, wherein the masking operation is associated with a diffuse attribute.
64. (Previously added) The method as recited in claim 57, wherein the masking operation is associated with a specular attribute.
65. (Previously added) A single-platform graphics pipeline system with an integrated masking operation, comprising:  
a transform module positioned on a single semiconductor platform for transforming graphics data;  
a lighting module positioned on the same single semiconductor platform as the transform module, the lighting module being for performing lighting operations on the graphics data;  
a set-up module positioned on the same single semiconductor platform as the transform module and the lighting module, the set-up module being for setting up the graphics data; and  
a rendering module positioned on the same single semiconductor platform as the transform module, the lighting module, and the set-up module, the rendering module being for 3-D rendering of the graphics data;  
wherein the single semiconductor platform is adapted for coupling to a central processing unit for receiving instructions therefrom;  
wherein a masking operation is capable of being performed utilizing the single semiconductor platform;  
wherein the single semiconductor platform is capable of operating with an application program interface.
66. (Previously added) The system as recited in claim 65, wherein the masking operation includes a 2-bit write mask for providing control to a word level of a buffer address.
67. (Previously added) The system as recited in claim 65, wherein the masking operation includes individually masking a write operation to at least one register component such

that unmasked components are taken from the register and masked components are bypassed.

68. (Previously added) The system as recited in claim 65, wherein the masking operation masks an enabled bit of a control vector for allowing analysis of all remaining bits after execution of the masking operation.
69. (Previously added) The system as recited in claim 65, wherein the masking operation is associated with an ambient attribute.
70. (Previously added) The system as recited in claim 65, wherein the masking operation is associated with a diffuse attribute.
71. (Previously added) The system as recited in claim 65, wherein the masking operation is associated with a specular attribute.
72. (Previously added) A method for graphics processing, comprising:
  - transforming graphics data from a first space to a second space;
  - lighting the graphics data;
  - performing a masking operation on the graphics data;
  - setting up the graphics data; and
  - 3-D rendering the graphics data;wherein the graphics data is set up, transformed and lighted, and the masking operation is performed, on a single semiconductor platform;
  - wherein the single semiconductor platform also operates with an application program interface.
73. (Previously added) The method as recited in claim 72, wherein the masking operation includes a 2-bit write mask for providing control to a word level of a buffer address.
74. (Previously added) The method as recited in claim 72, wherein the masking operation includes individually masking a write operation to at least one register component such

that unmasked components are taken from the register and masked components are bypassed.

75. (Previously added) The method as recited in claim 72, wherein the masking operation masks an enabled bit of a control vector for allowing analysis of all remaining bits after execution of the masking operation.
76. (Previously added) The method as recited in claim 72, wherein the masking operation is associated with an ambient attribute.
77. (Previously added) The method as recited in claim 72, wherein the masking operation is associated with a diffuse attribute.
78. (Previously added) The method as recited in claim 72, wherein the masking operation is associated with a specular attribute.
79. (Previously added) The method as recited in claim 72, wherein the single semiconductor platform is adapted for coupling to a central processing unit for receiving instructions therefrom.
80. (Previously added) A single-platform graphics pipeline system with an integrated masking operation, comprising:
  - a transform module positioned on a single semiconductor platform for transforming graphics data;
  - a lighting module positioned on the same single semiconductor platform as the transform module, the lighting module being for performing lighting operations on the graphics data;
  - a set-up module positioned on the same single semiconductor platform as the transform module and the lighting module, the set-up module being for setting up the graphics data;
  - a rendering module positioned on the same single semiconductor platform as the transform module, the lighting module, and the set-up module, the rendering module being for 3-D rendering of the graphics data; and



memory positioned on the same single semiconductor platform as the transform module, the lighting module, the set-up module, and the render module for storing the graphics data;

wherein the single semiconductor platform is adapted for coupling to a central processing unit for receiving instructions therefrom;

wherein a masking operation is performed utilizing the single semiconductor platform;

wherein a scissor operation is performed utilizing the single semiconductor platform;

wherein a clipping operation is performed utilizing the single semiconductor platform;

wherein the graphics data is blended utilizing the single semiconductor platform for blending triangles represented by vertex data associated with the graphics data;

wherein a vertex fog operation is performed on the graphics data utilizing the single semiconductor platform;

wherein the single semiconductor platform operates with a Direct3D application program interface;

wherein the single semiconductor platform also operates with an OpenGL application program interface.